Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computer and System Sciences 72 (2006) 206–219

JOURNAL OF COMPUTER AND SYSTEM SCIENCES

www.elsevier.com/locate/jcss

Hardness of approximating the Shortest Vector Problem in high ℓ_p norms ☆

Subhash Khot

Department of Computer Science, Princeton University, Princeton, NJ 08544, USA

Received 22 April 2004; received in revised form 24 October 2004

Available online 7 November 2005

Abstract

We present a new hardness of approximation result for the Shortest Vector Problem in ℓ_p norm (denoted by SVP_p). Assuming NP $\not\subseteq$ ZPP, we show that for every $\varepsilon > 0$, there is a constant $p(\varepsilon)$ such that for all integers $p \geqslant p(\varepsilon)$, the problem SVP_p has no polynomial time approximation algorithm with approximation ratio $p^{1-\varepsilon}$.

Keywords: Computational complexity; Lattices; Shortest vector problem; Approximation algorithms; Hardness of approximation

1. Introduction

An n-dimensional lattice $\mathcal{L}$ is a set of vectors $\{\sum_{i=1}^{n} b_i v_i \mid b_i \in \mathbb{Z}\}$ where $v_1, v_2, \ldots, v_n \in \mathbb{R}^m$ is a set of linearly independent vectors called the basis for the lattice. The same lattice could have many bases. Given a basis for an n-dimensional lattice, the *Shortest Vector Problem* asks for the shortest non-zero vector in the lattice. The length of the vectors can be measured in any ℓ_p norm ($p \geqslant 1$) and the corresponding problem is denoted by SVP_p. This problem has a beautiful history and we present some

☆ A preliminary version of this paper appeared in Proceedings of the 44th IEEE Symposium on Foundations of Computer Science, 2003. This research was supported in part by Sanjeev Arora's NSF Awards 0205594, CCR 0098180 and the David and Lucile Packard Fellowship.

E-mail address: khot@cs.princeton.edu.

0022-0000/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcss.2005.07.002

of the results below. For a more comprehensive list of references and a thorough treatment of the subject, we refer to Micciancio and Goldwasser's book [32]. We also recommend Micciancio's PhD thesis [30] and an expository article by Kumar and Sivakumar [23].

The Shortest Vector Problem has been studied since the time of Gauss ([16], 1801) who gave an algorithm for SVP_2 in two dimensions. The general problem for arbitrary dimensions was formulated by Dirichlet in 1842. The theory of Geometry of Numbers by Minkowski [33] deals with the existence of shortest non-zero vectors in lattices. In a celebrated result, Lenstra et al. [28] gave a polynomial time algorithm for approximating SVP_2 within factor $2^{n/2}$. This algorithm has numerous applications, e.g. factoring rational polynomials [28], breaking knapsack-based codes [25], checking the solvability by radicals [26] and integer programming in a fixed number of variables [28,27,19]. Schnorr [36] improved the approximation factor to $2^{O(n(\log\log n)^2/\log n)}$. Since all ℓ_p norms are within factor $\sqrt{n}$ from the ℓ_2 norm, these algorithms give similar approximations for SVP_p for any p. It is a major open problem whether SVP has polynomial factor approximations that run in polynomial time. Exact computation of SVP_2 in exponential time is also investigated, see for instance [20,5].

In 1981, van Emde Boas [37] proved that SVP_∞ is NP-hard and conjectured that the same is true for any ℓ_p norm. However proving NP-hardness for any finite p (in particular $p = 2$) was an embarrassing open problem for a long time. A breakthrough result by Ajtai [3] in 1998 finally showed that SVP_2 is NP-hard under randomized reductions. This was improved to a hardness of approximation result by Cai and Nerurkar [12], achieving a hardness factor of $(1 + \frac{1}{n^\delta})$. Another breakthrough by Micciancio [31] showed that SVP_p is hard to approximate within some constant factor, specifically, factor $2^{1/p} - \delta$ for every $\delta > 0$.

Showing hardness of approximation results for SVP was greatly motivated by Ajtai's reduction [2] from worst-case hardness to average-case hardness and the subsequent construction of a lattice-based public key cryptosystem by Ajtai and Dwork [4]. Ajtai showed that if there is a randomized polynomial time algorithm for solving SVP_2 on a non-negligible fraction of lattices from a certain natural class of lattices, then there is a randomized polynomial time algorithm for approximating SVP_2 on *every* instance within some polynomial factor n^c. Ajtai–Dwork's work gave hope, for the first time, that cryptography could be based on the (conjectured) worst-case hardness of a problem. Their work implies that if n^c-approximation to SVP_2 is hard, then one can construct a secure cryptosystem. The constant c was noted to be 19 in [10], and brought down to $9 + \delta$ by Cai and Nerurkar [11] and then to $4 + \delta$ by Cai [10]. Recently, Regev [35] gave an alternate construction of a public key cryptosystem based on $n^{1.5}$-hardness of SVP_2 (actually these results use a variant of SVP_2 called unique-SVP_2). Unfortunately, there are barriers to showing such strong hardness results. In fact, showing factor n NP-hardness would imply that NP $=$ coNP [24] and showing factor $\sqrt{n/O(\log n)}$ NP-hardness would imply that coNP $\subseteq$ AM [17] (and the polynomial hierarchy collapses in both cases). Recently, Aharonov and Regev [1] showed that if approximating SVP_2 within factor $\sqrt{n}$ is NP-hard then NP $=$ coNP.

Another related problem that has received much attention is the Closest Vector Problem (denoted by CVP_p) : Given a lattice and a vector y, the problem is to find the lattice vector that is closest to y in ℓ_p norm. In spite of the apparent similarity between SVP and CVP, they turn out to be quite different problems. Indeed, CVP_p was shown to be NP-hard for all $p \geqslant 1$ by van Emde Boas [37]. Arora et al. [6] used the PCP machinery to show that approximating CVP_p (for all $p \geqslant 1$) within factor $2^{\log^{1-\delta} n}$ is hard unless NP $\subseteq$ DTIME$(2^{poly(\log n)})$. This was improved to a factor $n^{1/\log\log n}$ NP-hardness result by Dinur et al. [14]. Incidentally, SVP_∞ seems to behave very much like CVP_∞. Dinur [13] showed factor

$n^{1/\log\log n}$ NP-hardness for both these problems. Thus for SVP, the cases $p = \infty$ and $p < \infty$ seem to be qualitatively different.

Our result: In this paper, we obtain an improved hardness result for SVP_p for large (but finite) values of p. Specifically, we show that:

Theorem 1.1. *For every $\varepsilon > 0$, there is a constant $p(\varepsilon)$ such that for all integers $p \geqslant p(\varepsilon)$, there is no polynomial time approximation to* SVP_p *within ratio $p^{1-\varepsilon}$ provided* $\mathrm{NP} \nsubseteq \mathrm{ZPP}$.

This improves the hardness factor $2^{1/p} - \delta$ by Micciancio [31] for all large values of p. The result, however, is only asymptotic and says nothing about small values of p. The value $p(\varepsilon)$ depends on a non-explicit constant in Raz's Parallel Repetition Theorem [34].

Our reduction is a randomized reduction and so are the reductions of Ajtai [3] and Micciancio [31]. A randomized reduction that gives a hardness factor K has the following properties:

1. The reduction is polynomial time and it maps a SAT instance to an SVP_p instance, i.e. a lattice presented as a basis.
2. In the YES case (i.e. if the SAT instance is satisfiable), the lattice has a non-zero vector of length L with high probability.
3. In the NO case (i.e. if the SAT instance is unsatisfiable), the lattice has no non-zero vector of length $K \cdot L$ w.h.p.

Our reduction makes no error in the YES case, meaning Property (2) holds with probability 1. Therefore, our hardness result holds under the assumption that $\mathrm{NP} \nsubseteq \mathrm{coRP}$ which is equivalent to $\mathrm{NP} \nsubseteq \mathrm{ZPP}$. On the other hand, Ajtai and Micciancio's reductions make an error in the YES case, but make no error in the NO case. Hence their hardness results require a stronger assumption that $\mathrm{NP} \nsubseteq \mathrm{RP}$ (Micciancio, however, gives a deterministic reduction under a certain number-theoretic conjecture). Our reduction is much simpler and maybe easier to derandomize.

Overview of the paper: We prove Theorem 1.1 via a reduction from the Label Cover problem which is defined in Section 2.3. The reduction is quite straightforward and Section 3 gives the basic idea of the reduction. Section 4 gives the full reduction. In Section 2.2, we explain a powerful technique that has been useful in subsequent work on SVP (see [21]).

Recent progress on SVP: After a preliminary version of this paper appeared, Khot and Vishnoi [22] showed a factor $(4/3)^{1-45.7/p}$ hardness for SVP_p for $p \geqslant 46$ and a factor $(3/2)^{1-111.7/p}$ for $p \geqslant 112$. All hardness results for SVP_p for $1 < p < \infty$ were superceded by Khot's [21] recent results. He showed that for any $1 < p < \infty$, SVP_p is hard to approximate within any constant factor assuming $\mathrm{NP} \nsubseteq \mathrm{RP}$ and within factor $2^{(\log n)^{1/2-\varepsilon}}$ assuming $\mathrm{NP} \nsubseteq \mathrm{RPTIME}(2^{poly(\log n)})$. In both these papers, the technique introduced in Section 2.2 plays a crucial role.

2. Problem definition and techniques

We first define the Shortest Vector Problem in ℓ_p norm. It is defined in a different (but equivalent) manner, without any reference to lattices.

2.1. Problem definition

The problem SVP_p is defined as follows: Given a vector $\mathbf{x}$ of integer variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and linear forms $\{\phi_1, \phi_2, \ldots, \phi_m\}$ where

$$\phi_i(\mathbf{x}) = \sum_{j=1}^{n} a_{ij} x_j, \quad a_{ij} \in \mathbb{R}.$$

The goal is to find a non-zero integer vector $\mathbf{x}$ which minimizes the following objective function:

$$\mathrm{OBJ} = \sum_{i=1}^{m} |\phi_i(\mathbf{x})|^p.$$

Remarks. (1) Think of the basis vectors in the lattice as the columns of the matrix $\{a_{ij}\}$ and think of the integer variables x_i's as the (unknown) coefficients when the shortest vector is written as an integer linear combination of the basis vectors. (2) We actually want to minimize $\mathrm{OBJ}^{1/p}$. In order to show a factor k-hardness for SVP_p, it suffices to show a factor k^p-hardness for the above objective function.

2.2. A useful technique

In this section, we describe one of the techniques used in this paper. This is a simple but powerful technique which has proved crucial in subsequent work on SVP (see [21]).

A common problem encountered in showing hardness of SVP is the following: We desire a reduction from some starting NP-hard problem, say Label Cover, Set Cover or CVP. Usually, it is straightforward to construct a set of vectors $\{v_1, v_2, \ldots, v_m\}$ which one could potentially use as the basis vectors for an instance of SVP. If the starting NP-hard problem is a YES instance, then there is a non-zero integer linear combination $\sum_{i=1}^{m} x_i v_i$ with short length. This combination corresponds to a correct labeling to Label Cover or a solution to the Set Cover, depending on the problem we started with. The combination typically has the property that δm of the coefficients x_i's are non-zero. If the starting NP-hard problem is a NO instance, we would like to show that there is no non-zero integer linear combination with short length. However, typically it so happens that each of the vectors v_i itself is a short vector. Thus for any j, setting $x_j = 1$ and $x_i = 0$ for $i \neq j$ gives a short lattice vector. In general, setting *too few* of x_i's to non-zero values produces a short vector. We wish to somehow enforce the condition that many of the x_i's must be set to a non-zero value.

We do this by augmenting the vectors v_i's by one extra co-ordinate a_i. Call these augmented vectors $\tilde{v}_i = (v_i, a_i)$ and let them be the basis vectors for an SVP instance. The set of integers $\{a_i : 1 \leqslant i \leqslant m\}$ satisfies:

- For any set $Y \subseteq [m]$, $|Y| = \delta m$, the integers $\{a_j \mid j \in Y\}$ have a non-zero $\{0, 1, -1\}$-linear combination that vanishes.
- For any set $Z \subseteq [m]$, $|Z| < \frac{\delta m}{20 \log(1/\delta)}$, a non-zero $\{0, 1, -1\}$-linear combination of integers $\{a_j \mid j \in Z\}$ cannot vanish.

It can be shown that choosing m random integers from the range $[1, 2, \ldots, 2^{\delta m/2}]$ satisfies these properties with high probability (the first property is satisfied with probability 1 using the Pigeon-Hole Principle).

In the YES case, since δm of the x_i's are non-zero, one could hope to set them to appropriate $\{0, 1, -1\}$ values so that $\sum_{i=1}^{m} x_i a_i = 0$ and the vector $\sum_{i=1}^{m} x_i v_i$ is short.

In the NO case, assume for the moment that x_i's are restricted to take values $\{0, 1, -1\}$. It is clear that if at most $\frac{\delta m}{20 \log(1/\delta)}$ of the x_i's are non-zero, then $\sum_{i=1}^{m} x_i a_i$ cannot vanish. One can apply a huge penalty if this sum (which is the last co-ordinate of the linear combination $\sum_{i=1}^{m} x_i \tilde{v}_i$) does not vanish. Thus, we are able to enforce the constraint that one must set at least $\frac{\delta m}{20 \log(1/\delta)}$ of the x_i's to non-zero value. In general, x_i's could take arbitrary integer values (not just $\{0, 1, -1\}$), but this can be handled as well, as we will see.

Construction of the set $\{a_i | 1 \leqslant i \leqslant m\}$ is the only place where our reduction is randomized. We would like to remark that Micciancio [31] also needs a gadget to enforce a similar condition and almost all work in his paper is devoted to constructing this gadget. More specifically, his gadget is a sophisticated lattice and he needs to enforce the condition that *one* particular coefficient x_{i_0} in the integer linear combination is set to a non-zero value.

Our reduction is from a problem called "Label Cover" which we define next. This problem was introduced by Arora et al. [6] and can be equivalently viewed as the problem of finding good prover strategies in a 2-prover 1-round game.

2.3. The Label Cover problem

The Label Cover problem is defined as follows: We are given a bipartite graph $G = (V, W, E)$ with left-side vertices V, right-side vertices W and a set of edges E. The goal is to assign one "label" to every vertex, where the vertices in W are required to receive a label from set $[R]$ and the vertices in V are required to receive a label from set $[S]$. Thus, a labeling A is just a map $A : W \mapsto [R]$, $A : V \mapsto [S]$. The labeling is supposed to satisfy certain constraints given by maps $\pi_{v,w} : [R] \mapsto [S]$. There is one such map for every edge $(v, w) \in E$. A labeling A "satisfies" an edge (v, w), if

$$\pi_{v,w}(A(w)) = A(v).$$

The optimum OPT of the label cover problem is defined to be the maximum fraction of edges satisfied by any labeling. Let $n = |V|$, $m = |W|$. Let D be the degree of every vertex in V. Think of n, m as growing and $m \geqslant n$. Think of R, S, D as constants and $R \geqslant D \geqslant S$.

The following theorem can be obtained by combining the PCP Theorem [8,7] with Raz's Parallel Repetition Theorem [34]. This theorem is the starting point for most of the recent PCP constructions and hardness results (e.g. [18]). We sketch a proof for completeness' sake.

Theorem 2.1. *There exists an absolute constant $\gamma > 0$ such that for every sufficiently large integer R, it is NP-hard to distinguish whether a Label Cover problem specified as*

$$(G(V, W, E), n, m, \{\pi_{v,w}\}, [R], [S], D)$$

has OPT $= 1$ or OPT $< \frac{1}{R^{\gamma}}$.

It can be ensured that for an integer parameter u, $R = 6^u$, $S = 3^u$, $D = 5^u$, $m = (5/2)^u n$. Also, for every edge (v, w), the map $\pi_{v,w} : [R] \mapsto [S]$ is "regular", meaning for every $j \in [S]$, there are exactly R/S elements in $[R]$ that are mapped to j.

Proof. As shown in [15], there is $\theta_0 > 0$ such that it is NP-hard to tell whether a 5-regular graph is 3-colorable or no coloring of its vertices with 3 colors can make $1 - \theta_0$ fraction of the edges non-monochromatic. Starting with this gap-problem, one can construct a 2-Prover-1-Round game as follows: Pick a random edge (x, y) of the graph and one of its endpoints at random (say x). Ask prover P_2 for coloring of the vertices $\{x, y\}$; his answer is supposed to be one of the 6 valid (non-monochromatic) colorings. Ask prover P_1 for the color of x; his answer is supposed to be one of the 3 colors. Accept if and only if the answers of P_1 and P_2 agree on color of x. It is easy to see that this game has perfect completeness and soundness strictly less than 1. Applying Parallel Repetition Theorem gives an instance of Label Cover with all the properties listed above. □

Remark. The hardness factor for SVP_p that we will achieve is $k^{1-\varepsilon}$. We will set $k = R^{\gamma/20}$ and $p = O(k)$ where R is as in Theorem 2.1.

3. The basic idea in the reduction

Here we describe the basic idea of the reduction. One needs to fix a lot of technical details later and we present a complete reduction in Section 4.

Theorem 2.1 gives an instance of the Label Cover problem specified as

$$(G(V, W, E), n, m, \{\pi_{v,w}\}, [R], [S], D).$$

The integer variables of the SVP_p will be

$$\{x_{w,i} \mid w \in W, i \in [R]\}.$$

For a fixed w, let $B(w)$ be the "block" of variables defined as

$$B(w) = \{x_{w,i} \mid i \in [R]\}.$$

For a vertex $v \in V$, let $N(v) \subseteq W$ denote the set of neighbors of v with $|N(v)| = D$. There will be one linear form in SVP_p for every $v \in V$ and every sequence $(w_1, w_2, \ldots, w_S)$ where $w_j \in N(v)$ for $1 \leqslant j \leqslant S$. The linear form is the sum of R variables, with R/S variables each from the block $B(w_j)$ (we use the regularity of the Label Cover instance). The linear form is defined to be

$$\sum_{j=1}^{S} \sum_{i \in [R]: \pi_{v,w_j}(i) = j} x_{w_j,i}. \tag{1}$$

Note that the total number of linear forms is nD^S.

3.1. Completeness

We will show that if there is a labeling A for the label cover problem satisfying every edge (i.e. $\mathrm{OPT} = 1$), then the problem SVP_p has a solution with the objective function $\mathrm{OBJ} = nD^S$. Let $A : W \mapsto [R]$, $A : V \mapsto [S]$ be such a labeling. For every edge (v, w) in the Label Cover instance, we have $\pi_{v,w}(A(w)) = A(v)$.

Define a solution as

$$x_{w,i} = \begin{cases} 1 & \text{if } A(w) = i, \\ 0 & \text{otherwise.} \end{cases}$$

Note that there are nD^S linear forms of type (1). We will show that each of these linear forms equals 1. Indeed, out of all the variables in the linear form (1), exactly one equals 1 and the rest are all 0. The non-zero variable is $x_{w_j,i}$ for which $j = A(v)$, $i = A(w_j)$.

3.2. Soundness

We wish to show that if OPT $< 1/R^\gamma$ for the Label Cover problem, then for any non-zero integer vector $\mathbf{x} = \{x_{w,i}\}$, the objective function OBJ is at least $\frac{1}{2}nD^{S-k}k^p$. We will show this only for a restricted class of vectors, i.e. vectors $\mathbf{x} = \{x_{w,i}\}$ which "arise" out of labelings $A : W \mapsto [R]$. Eventually we want this to work for *every* non-zero vector $\{x_{w,i}\}$. This involves a lot of technical details and we present the full reduction in Section 4.

So let us assume $A : W \mapsto [R]$ is an assignment and $\{x_{w,i}\}$ is the corresponding vector, i.e.

$$x_{w,i} = \begin{cases} 1 & \text{if } A(w) = i, \\ 0 & \text{otherwise.} \end{cases}$$

For every vertex $v \in V$, define a set of labels $\Psi(v) \subseteq [S]$ as follows:

$$\Psi(v) = \{\pi_{v,w}(A(w)) \mid w \in N(v)\}.$$

Lemma 3.1. *Let $k = R^{\gamma/20}$. For at least half the vertices in V, $|\Psi(v)| \geqslant k$.*

Proof. Assume on the contrary that half the vertices in V have $|\Psi(v)| \leqslant k$. Define label for vertex v to be a random element of $\Psi(v)$. Note that for every $w \in N(v)$, $\pi_{v,w}(A(w)) \in \Psi(v)$. Therefore with probability $1/k$, the edge (v, w) is satisfied by this random labeling to v. Hence there exists a labeling for the Label Cover problem that satisfies at least a fraction $\frac{1}{2k}$ of the edges. This is a contradiction since $\frac{1}{2k} > \frac{1}{R^\gamma}$. □

Call the vertices satisfying $|\Psi(v)| \geqslant k$ as "good". For every good vertex v, assume w.l.o.g. that $\{1, 2, \ldots, k\} \subseteq \Psi(v)$. Thus for every $1 \leqslant j \leqslant k$, there exists $w_j^* \in N(v)$ such that $\pi_{v,w_j^*}(A(w_j^*)) = j$. Hence for any sequence

$$(w_1^*, w_2^*, \ldots, w_k^*, w_{k+1}, \ldots, w_S),$$

where $w_{k+1}, \ldots, w_S \in N(v)$ are arbitrary, the linear form (1) is at least equal to k. This contributes k^p to the objective function OBJ (we are working in ℓ_p norm).

Half the vertices are good, hence OBJ $\geqslant \frac{n}{2}D^{S-k}k^p$.

3.3. Hardness factor

Note that the objective function is nD^S in completeness case and at least $\frac{1}{2}nD^{S-k}k^p$ in the soundness case. Choose p such that

$$\tfrac{1}{2}nD^{S-k}k^p > nD^S k^{(1-\varepsilon)p}.$$

Thus there is a penalty of a factor $k^{(1-\varepsilon)p}$ in the soundness case. Noting that $D \leqslant R$, $k = R^{\gamma/20}$, we see that it suffices to take $p = \frac{20k}{\varepsilon\gamma}$. This gives hardness factor of $k^{1-\varepsilon}$ or $p^{1-\varepsilon'}$ as desired. This completes the basic idea of the reduction. We give the full reduction in the next section.

4. Full reduction

The set of integer variables is the same, namely

$$\mathbf{x} = \{x_{w,i} \mid w \in W, i \in [R]\}.$$

There will be four types of linear forms. These forms are supposed to "handle" different types of non-zero vectors $\mathbf{x}$ in the soundness case. The exact role of these linear forms will be clear as we go along.

Type-1 linear forms:

$$\forall\, w \in W,\quad \forall i \in [R],\qquad x_{w,i}.$$

Thus the Type-1 linear forms are just the variables themselves.

Type-2 linear forms:

$$\sum_{w \in W, i \in [R]} a_{w,i} x_{w,i}.$$

Thus there is only one Type-2 linear form. The coefficients $a_{w,i}$ in this linear form are randomly chosen integers from the range $[1, 2, \ldots, 2^{m/2}]$.

Type-3 linear forms:

$$\forall\, w \in W,\quad \sum_{i=1}^{R} \pm x_{w,i}.$$

Thus there are 2^R Type-3 linear forms for every $w \in W$. There are R variables in every form and there is one linear form for every choice of $+/-$ sign.

Type-4 linear forms:

$$\forall\, v \in V,\quad \forall w_1, w_2, \ldots, w_S \in N(v),\qquad \sum_{j=1}^{S} \sum_{i \in [R]: \pi_{v,w_j}(i) = j} \pm x_{w_j,i}.$$

There are $2^R D^S$ linear forms for every $v \in V$. These are essentially the linear forms used in Section 3, except that now we take all the $+/-$ combinations. Note that there are R variables in each form.

In the completeness case, Type-2 form will contribute 0. Type-1, Type-3 and Type-4 will contribute (at most) m, $2^R m$ and $2^R n D^S$, respectively. We multiply Type-1, Type-3 and Type-4 forms by appropriate "balancing" quantities C_1, C_3, C_4 so that they contribute equally towards the objective function. More precisely, C_1, C_3, C_4 are chosen so that

$$C_1^p\, m = C_3^p\, 2^R m = C_4^p\, 2^R n D^S.$$

We multiply Type-2 form by a huge quantity. Thus we incur a huge penalty unless the Type-2 form vanishes (it will vanish in the completeness case, so we are fine).

In the soundness case, we will show that any non-zero vector $\mathbf{x} = \{x_{w,i}\}$ either produces a non-zero value in Type-2 form or it pays a penalty of factor $k^{(1-\varepsilon)p}$ for at least one of the remaining three types.

4.1. Completeness

In the completeness case, let $A : W \mapsto [R]$, $A : V \mapsto [S]$ be a correct labeling. Let

$$x_{w,i} = \begin{cases} 0, 1 \text{ or } -1 & \text{if } A(w) = i, \\ 0 & \text{otherwise.} \end{cases}$$

The choice of 0, 1, -1 for the variables $\{x_{w,A(w)}\}$ is made such that Type-2 form vanishes. To make this choice, we use the Pigeon-Hole principle. Consider the set of m variables $\{x_{w,A(w)}\}$ and the corresponding coefficients $a_{w,A(w)}$ in Type-2 form. Consider the 2^m different sums for all subsets of these m coefficients. These sums take integer values in the range $[1, 2, 3, \ldots, m2^{m/2}]$. Hence sums for two distinct subsets must be equal, which gives a vanishing $\{0, 1, -1\}$ linear combination of the integers $a_{w,A(w)}$.

Now we look at the remaining three types.

- For Type-1 forms, note that at most m of the variables are ± 1, the rest are 0. So we get contribution of at most m (times the balancing factor C_1^p which we hide).
- For Type-3 forms, note that for every w, there is at most one variable $x_{w,i}$ that is ± 1. Thus we get contribution of at most $2^R m$ (times the balancing factor C_3^p which we hide).
- Every Type-4 form is ± 1 as seen in Section 3.1 and it might even be 0 since we "turn off" some of the variables $x_{w,A(w)}$ to 0. Thus Type-4 forms contribute at most $2^R n D^S$ (times the balancing factor C_4^p which we hide).

5. Soundness

The crux of the soundness analysis is as in Section 3.2. However, we have to handle cases when $x_{w,i}$'s are negative, or very few of them are non-zero. We do this in several stages.

For a vector $\mathbf{x} = \{x_{w,i}\}$, let $\#\mathbf{x}$ denote the number of variables (or coordinates) that are non-zero. For a block of variables $B(w)$, let $\#B(w)$ denote the number of non-zero variables in this block.

5.1. Handling $\mathbf{x}$ with $\#\mathbf{x} \leqslant \frac{m}{20 \log R}$ and $\|\mathbf{x}\|_1 \geqslant mR$

Such $\mathbf{x}$ are handled by Type-1 forms. Note that when at most $\frac{m}{20 \log R}$ coordinates are non-zero and the ℓ_1 norm is at least mR, then ℓ_p norm is minimized when all the non-zero coordinates are equal to $\frac{mR}{m/(20 \log R)}$. Hence

$$\sum_{w,i} |x_{w,i}|^p \geqslant (20R \log R)^p \frac{m}{20 \log R} \geqslant R^p m \geqslant k^p m.$$

Note that the contribution of Type-1 forms in the completeness case is at most m. Thus we get a penalty of factor k^p for Type-1 forms.

5.2. *Handling* $\mathbf{x}$ *with* $\#\mathbf{x} \leqslant \frac{m}{20 \log R}$ *and* $\|\mathbf{x}\|_1 \leqslant mR$

These are handled by Type-2 linear form. We show that with high probability, Type-2 linear form *does not* vanish for any such $\mathbf{x}$.

The coefficients in this linear form are randomly chosen integers from the range $[1, 2, 3, \ldots, 2^{m/2}]$. Hence for any non-zero vector $\mathbf{x}$, the probability that Type-2 form vanishes is at most $\frac{1}{2^{m/2}}$. We count the number of vectors $\mathbf{x}$ such that $\#\mathbf{x} \leqslant \frac{m}{20 \log R}$ and $\|\mathbf{x}\|_1 \leqslant mR$. We show that there are not too many of them and we can take a union bound. Number of such $\mathbf{x}$'s can be bounded by

$$\binom{mR}{\frac{m}{20 \log R}} \cdot 2^{m/(20 \log R)} \cdot \left(\#\text{ non-negative integer solutions to }: y_1 + y_2 + \ldots + y_{\frac{m}{20 \log R}} \leqslant mR\right)$$
$$\leqslant \binom{mR}{\frac{m}{20 \log R}} \cdot 2^{m/(20 \log R)}\, mR \cdot \binom{2mR}{\frac{m}{20 \log R}} \leqslant 2^{2m/5} \ll 2^{m/2}.$$

Here we used the fact that

$$\binom{M}{\delta M} \leqslant 2^{H(\delta)M} \leqslant 2^{2\delta \log(1/\delta) M}.$$

Thus from now onwards, we assume that $\#\mathbf{x} \geqslant \frac{m}{20 \log R}$.

5.3. *Avoiding the problem of negative or large values*

In general, the variables $x_{w,i}$ could be positive or negative and could take large integer values. This is handled by averaging over all the $+/-$ linear combinations and this is the reason why Type-3 and Type-4 forms appear with all possible $+/-$ combinations.

Lemma 5.1. *Let $x_1, x_2, \ldots, x_k$ be k non-zero integers. Let $a_i \in \{1, -1\}$ be chosen randomly. Let p be an integer. Then*

$$E_{a_1, a_2, \ldots, a_k}\left[\left|\sum_{i=1}^{k} a_i x_i\right|^p\right] \geqslant \max\{k^{p-1-k}, k^{(p-1)/2}\}.$$

Proof. Assume first that p is even. We can expand out the product $(\sum_{i=1}^{k} a_i x_i)^p$ and take the expectation of each term separately. For the terms in which some $a_i x_i$ occurs to an odd power, the expectation is zero. For the remaining terms, the expectation is at least 1. Thus the expectation is at least the number of terms such that every $a_i x_i$ occurs to an even power. In other words, we want to count the number of functions $f : [p] \mapsto [k]$ such that every $j \in [k]$ has even number of pre-images. Considering functions where $f(1) = f(2), f(3) = f(4), \ldots, f(p-1) = f(p)$, we get a bound of $k^{p/2}$. Another way is to take an arbitrary function $g : [p-k] \mapsto [k]$ and then "extend" it to a function $\tilde{g} : [p] \mapsto [k]$ where the values $\tilde{g}(p-k+1), \tilde{g}(p-k+2), \ldots, \tilde{g}(p)$ are chosen to make sure that for $\tilde{g}$, every $j \in [k]$ has an even number of pre-images. This gives a bound of k^{p-k}.

When p is odd, we can use the inequality

$$E_{a_1,a_2,\ldots,a_k}\left[\left|\sum_{i=1}^{k} a_i x_i\right|^p\right] \geqslant E_{a_1,a_2,\ldots,a_k}\left[\left|\sum_{i=1}^{k} a_i x_i\right|^{p-1}\right]$$

and then use the previous argument for $p-1$. $\square$

Lemma 5.1 implies that for any set of R variables, with at least k of them non-zero, when summed over all $+/-$ combinations, the contribution to the objective function OBJ is at least $2^R \max\{k^{p-1-k}, k^{(p-1)/2}\}$.

5.4. Handling one more annoying case

One more annoying case is when most of the non-zero variables belong to blocks $B(w)$ such that these blocks themselves have too many non-zero variables. To be precise, we want to avoid the situation where

$$\sum_{B(w):\#B(w)\geqslant k^3} \#B(w) \geqslant \frac{m}{40 \log R}. \tag{2}$$

This case is handled by Type-3 linear forms. By Lemma 5.1, for any block $B(w)$, the contribution of Type-3 forms towards the objective function is at least $2^R(\#B(w))^{(p-1)/2}$. Hence the contribution of blocks in (2) is at least,

$$\sum_{B(w):\#B(w)\geqslant k^3} 2^R(\#B(w))^{(p-1)/2}$$

which is minimized when all $\#B(w)$ are equal to k^3 and there are $\frac{m}{k^3 40 \log R}$ of them. Thus the contribution is at least

$$2^R(k^3)^{(p-1)/2}\frac{m}{k^3 40 \log R} \geqslant k^p 2^R m.$$

Note that the contribution in the completeness case is $2^R m$ and therefore one gets a penalty of factor k^p as desired.

5.5. Finishing the proof

After handling all the annoying cases, we can now assume that $\#\mathbf{x} \geqslant \frac{m}{20 \log R}$ and that

$$\sum_{B(w):\#B(w)\geqslant k^3} \#B(w) \leqslant \frac{m}{40 \log R}.$$

This implies that for at least $\frac{m}{k^3 40 \log R} = \delta m$ vertices $w \in W$, the block $B(w)$ contains at least one non-zero variable. Let

$$W' = \{w \mid B(w) \text{ contains at least one non-zero variable}\}.$$

We have $|W'| \geqslant \delta |W|$. Fix one non-zero variable in $B(w)$ for every $w \in W'$ and let this variable be $x_{w,A(w)}$ (thus we get an assignment A of labels to vertices in W').

By an averaging argument, for at least $\delta/4$ fraction of the vertices $v \in V$, at least $\delta/4$ fraction of their neighbors are in W'. Call any such vertex v "good". For any good vertex v, let

$$\Psi(v) = \{\pi_{v,w}(A(w)) \mid w \in W', w \in N(v)\}.$$

Lemma 5.2. *For at least half the good vertices v, $|\Psi(v)| \geqslant k$.*

Proof. Assume on the contrary that for half of the good vertices v, $|\Psi(v)| \leqslant k$. Thus for every such vertex v, we can assign at most k labels such that for every neighbor $w \in W'$ of v, the label $\pi_{v,w}(A(w))$ is included. Choosing at random, one of the at most k labels for every such vertex v gives a labeling to the Label Cover problem that satisfies following fraction of edges:

$$\frac{\delta}{8}\frac{\delta}{4}\frac{1}{k} = \frac{1}{32 \cdot 1600(\log R)^2 k^7} \geqslant \frac{1}{R^\gamma} \quad \text{since } k = R^{\gamma/20}.$$

This contradicts Theorem 2.1. □

Hence we assume that for at least half of the good vertices v, $|\Psi(v)| \geqslant k$. For any such vertex v, assume w.l.o.g. that $\{1, 2, \ldots, k\} \subseteq \Psi(v)$. Thus for every $1 \leqslant j \leqslant k$, there exists $w_j^* \in N(v)$ such that $\pi_{v,w_j^*}(A(w_j^*)) = j$. Hence for any sequence $(w_1^*, w_2^*, \ldots, w_k^*, w_{k+1}, \ldots, w_S)$ where $w_{k+1}, \ldots, w_S \in N(v)$ are arbitrary, the Type-4 linear form has at least k non-zero variables. Applying Lemma 5.1, we get a contribution of at least $2^R D^{S-k} k^{p-1-k}$ towards the objective function.

$\delta/4$ fraction of the vertices v are good, hence the objective function is at least

$$\left(\frac{\delta}{8} n\right) 2^R D^{S-k} k^{p-1-k}.$$

Note that the contribution in the completeness case is at most $2^R n D^S$. We will choose p such that

$$\frac{\delta}{8} n 2^R D^{S-k} k^{p-1-k} \; > \; 2^R n D^S \cdot k^{(1-\varepsilon)p}.$$

Thus we get a penalty of factor $k^{(1-\varepsilon)p}$ in the soundness case. Noting that $D \leqslant R$, $k = R^{\gamma/20}$, we see that it suffices to take $p = \frac{20k}{\varepsilon\gamma}$. This gives hardness factor of $k^{1-\varepsilon}$ or $p^{1-\varepsilon'}$ as desired. This completes the full reduction.

Note that the reduction has the following property: If the Label Cover instance is a YES instance, then the SVP_p instance always has a vector of length say L. If the Label Cover instance is a NO instance, then the lattice has no vector of length $p^{1-\varepsilon'} L$ w.h.p. Therefore, this is a coRP-reduction. In other words, if there were a polynomial time approximation algorithm for SVP_p with factor $p^{1-\varepsilon'}$, then NP $\subseteq$ coRP which implies NP $\subseteq$ ZPP.

Remark. It is crucial that the soundness parameter of the Label Cover problem is $1/R^\gamma$, i.e. polynomially small in the domain size R. Thus we use Raz's Parallel Repetition Theorem in a very strong sense, namely, the error goes down exponentially with the number of repetitions.

Acknowledgements

Many thanks to Sanjeev Arora for suggesting the use of Pigeon-Hole principle. Thanks to Daniele Micciancio for pointing out that the reduction in this paper is a ZPP-reduction. Thanks also to Oded Regev, Miki Ajtai, Amit Chakrabarti, Ravi Kumar and Siva Kumar for reading a preliminary write-up of this paper and for their excellent comments.

References

[1] D. Aharonov, O. Regev, Lattice problems in NP $\cap$ coNP, in: Proceedings of the 45th IEEE Symposium on Foundations of Computer Science, 2004, pp. 362–371.
[2] M. Ajtai, Generating hard instances of lattice problems, in: Proceedings of the 28th ACM Symposium on the Theory of Computing, 1996, pp. 99–108.
[3] M. Ajtai, The shortest vector problem in L_2 is NP-hard for randomized reductions, in: Proceedings of the 30th ACM Symposium on the Theory of Computing, 1998, pp. 10–19.
[4] M. Ajtai, C. Dwork, A public-key cryptosystem with worst-case/average-case equivalence, in: Proceedings of the 29th ACM Symposium on the Theory of Computing, 1997, pp. 284–293.
[5] M. Ajtai, R. Kumar, D. Sivakumar, A sieve algorithm for the shortest lattice vector problem, in: Proceedings of the 33rd ACM Symposium on the Theory of Computing, 2001, pp. 601–610.
[6] S. Arora, L. Babai, J. Stern, E.Z. Sweedyk, The hardness of approximate optima in lattices, codes and systems of linear equations, J. Comput. System Sci. 54 (1997) 317–331.
[7] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy, Proof verification and the hardness of approximation problems, J. ACM 45 (3) (1998) 501–555.
[8] S. Arora, S. Safra, Probabilistic checking of proofs: a new characterization of NP, J. ACM 45 (1) (1998) 70–122.
[10] J.Y. Cai, Applications of a new transference theorem to Ajtai's connection factor, Discrete Appl. Math. 126 (1) (2003) 9–31.
[11] J.Y. Cai, A. Nerurkar, An improved worst-case to average-case connection for lattice problems, in: Proceedings of the 38th IEEE Symposium on Foundations of Computer Science, 1997.
[12] J.Y. Cai, A. Nerurkar, Approximating the SVP to within a factor $(1 + 1/\dim^{\varepsilon})$ is NP-hard under randomized reductions, J. Comput. System Sci. 59 (2) (1999) 221–239.
[13] I. Dinur, Approximating SVP_∞ to within almost polynomial factors is NP-hard, Combinatorica 23 (2) (2003) 205–243.
[14] I. Dinur, G. Kindler, R. Raz, S. Safra, Approximating CVP to within almost-polynomial factors is NP-hard, Combinatorica 23 (2) (2003) 205–243.
[15] U. Feige, M. Halldorsson, G. Kortsarz, Approximating the domatic number, in: Proceedings of the 32nd Symposium on the Theory of Computing, 2000, pp. 134–143.
[16] C.F. Gauss, Disquisitiones arithmeticae. (leipzig 1801), art. 171, Yale University Press, 1966. (English translation by A.A. Clarke).
[17] O. Goldreich, S. Goldwasser, On the limits of non-approximability of lattice problems, J. Comput. System Sci. 60 (3) (2000) 540–563.
[18] J. Håstad, Some optimal inapproximability results, J. ACM 48 (2001) 798–859.
[19] R. Kannan, Improved algorithms for integer programming and related lattice problems, in: Proceedings of the 15th ACM Symposium on Theory of Computing, 1983, pp. 193–206.
[20] R. Kannan, Minkowski's convex body theorem and integer programming, Math. Oper. Res. 12 (1987) 415–440.
[21] S. Khot, Hardness of approximating the shortest vector problem in lattices, in: Proceedings of the 45th IEEE Symposium on Foundations of Computer Science, 2004, pp. 126–135.
[22] S. Khot, N. Vishnoi, Hardness of lattice problems in ℓ_p norm, manuscript, 2004.
[23] R. Kumar, D. Sivakumar, Complexity of SVP—a reader's digest, SIGACT News, vol. 32(3), in: L. Hemaspaandra (Ed.), Complexity Theory Column, 2001, pp. 40–52.
[24] J. Lagarias, H. Lenstra, C. Schnorr, Korkine–Zolotarev bases and successive minima of a lattice and its reciprocal lattice, Combinatorica 10 (1990) 333–348.

[25] J. Lagarias, A. Odlyzko, Solving low-density subset sum problems, J. ACM 32 (1) (1985) 229–246.
[26] S. Landau, G. Miller, Solvability of radicals is in polynomial time, J. Comput. System Sci. 30 (2) (1985) 179–208.
[27] H. Lenstra, Integer programming with a fixed number of variables, Technical Report 81-03, University of Amsterdam, Amsterdam, 1981.
[28] A. Lenstra, H. Lenstra, L. Lovász, Factoring polynomials with rational coefficients, Math. Ann. 261 (1982) 513–534.
[30] D. Micciancio, Ph.D. Thesis, MIT, 1998.
[31] D. Micciancio, The shortest vector problem is NP-hard to approximate to within some constant, SIAM J. Comput. 30 (6) (2001) 2008–2035.
[32] D. Micciancio, S. Goldwasser, Complexity of Lattice Problems, a Cryptographic Perspective, Kluwer Academic Publishers, Dordrecht, 2002.
[33] H. Minkowski, Geometrie der zahlen, Leizpig, Tuebner, 1910.
[34] R. Raz, A parallel repetition theorem, SIAM J. Comput. 27 (3) (1998) 763–803.
[35] O. Regev, New lattice based cryptographic constructions, in: Proceedings of the 35th ACM Symposium on the Theory of Computing, 2003.
[36] C.P. Schnorr, A hierarchy of polynomial-time basis reduction algorithms, Theoret. Comput. Sci. 53 (2–3) (1987) 201–224.
[37] P. van Emde Boas, Another NP-complete problem and the complexity of computing short vectors in a lattice, Technical Report 81-04, Mathematische Instiut, University of Amsterdam, Amsterdam, 1981.